Problems Related to Classical and Universal List Broadcasting

By: Saber Gholami
Supervisor: Professor Hovhannes Harutyunyan

Concordia University,
Department of Computer Science and Software Engineering

Nov 9^{th}, 2022

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Introduction

- Growth of using computer networks,
- Great attention to all major problems in this area,
- Information dissemination,
- Broadcasting:
  - Process of distributing a message starting from a single node (*originator*) to all other nodes of the network using the network's links.

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.
- $B_{cl}(u, G)$: minimum time required to finish the broadcasting from u.
- $B_{cl}(G) = \max\{B_{cl}(u, G) | u \in V(G)\}$

Preliminaries

- The network: $G = (V, E)$, originator $u \in V$.
- $B_{cl}(u, G)$: minimum time required to finish the broadcasting from u.
- $B_{cl}(G) = \max\{B_{cl}(u, G) | u \in V(G)\}$
- Two major problems in this area:
  - ◇ Broadcast time problem,
  - ◇ Network design.

Literature Review - Broadcast time problem

- Finding $B_{cl}(u, G)$ or $B_{cl}(G)$,
- Broadcast scheme: ordering of the neighbours of each vertex, depending on the originator:
  - u: originator,
  - once v gets informed, it will follow its list L_v^u,
  - Each vertex has to maintain up to $|V|$ different lists and know the originator to perform broadcasting.

[1] Peter J. Slater, Ernest J. Cockayne, and Stephen T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing, 10(4):692–701, 1981.*

Literature Review - Broadcast time problem

- Finding $B_{cl}(u, G)$ or $B_{cl}(G)$,
- Broadcast scheme: ordering of the neighbours of each vertex, depending on the originator:
  - u: originator,
  - once v gets informed, it will follow its list L_v^u,
  - Each vertex has to maintain up to $|V|$ different lists and know the originator to perform broadcasting.
- NP-Hard in arbitrary graphs [1],
- Directions to follow:
  - Exact solution for a specific graph,
  - Heuristic,
  - Approximation algorithms.

[1] Peter J. Slater, Ernest J. Cockayne, and Stephen T. Hedetniemi. Information dissemination in trees. *SIAM Journal on Computing, 10(4):692–701, 1981.*

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - Each vertex v has a single list L_v to follow, regardless of the originator.

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - Each vertex v has a single list L_v to follow, regardless of the originator.
- Two sub-models:
  - **Non-adaptive $B_{na}(G)$:** send to all vertices on the list,
  - **Adaptive $B_a(G)$:** skip the vertices from which the message is received.

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Broadcast time problem - cont.

- Broadcasting with universal lists:
  - Each vertex v has a single list L_v to follow, regardless of the originator.
- Two sub-models:
  - **Non-adaptive $B_{na}(G)$:** send to all vertices on the list,
  - **Adaptive $B_a(G)$:** skip the vertices from which the message is received.
- Introduced indirectly by Slater et al. [1]; for any Tree, $B_{cl}(T) = B_a(T)$.
- Diks and Pelc [2] distinguished between adaptive and non-adaptive models,
  - Also proposed several broadcast schemes for different graphs
- The hardness of the problem is unknown.

[1] Slater, P.J., Cockayne, E.J. and Hedetniemi, S.T., 1981. Information dissemination in trees. *SIAM Journal on Computing*, 10(4), pp.692-701..
[2] Diks, K. and Pelc, A., 1996. Broadcasting with universal lists. *Networks*, 27(3), pp.183-196.

Literature Review - Network Design

- Graph G on n vertices is a broadcast graph *(bg)* under classical model if $B_{cl}(G) = \lceil \log n \rceil$,
- A *bg* with minimum number of edges is called a minimum broadcast graph *(mbg)*,
- The number of edges of an *mbg* on n vertices: $B(n)$ or $B^{(cl)}(n)$.

Literature Review - Network Design - cont.

- $B^{(cl)}(n)$ is known for very few n,
- Exact values:
  - ◇ $n \leq 32$, except for $23, 24, 25$.
  - ◇ $n = 2^k$, Hypercubes | Knödel Graph | Recursive circulant graph
  - ◇ $n = 2^k - 2$, Knödel Graph
- Several upper bounds and lower bounds,
- No result under the universal lists model.

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Fully Connected Trees

- A Fully Connected Tree FCT_n:
  - ◇ A Clique of size n +
  - ◇ n arbitrary trees.
- Previous result on classical model: An algorithm with a time complexity of $O(|V| \log |V|)$ [1]

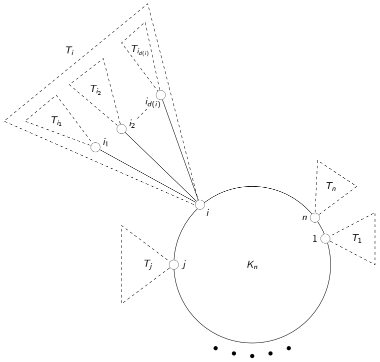


Figure: A Fully Connected Tree FCT_n

[1] Harutyunyan, H. A., Maraachlian, E. (2009a). Broadcasting in Fully Connected Trees. In 15th IEEE International Conference on Parallel and Distributed Systems, (ICPADS) (pp. 740–745).

FCT_n - Broadcast Algorithm for Root Vertices

- Instead of finding $B_{cl}(i, FCT_n)$, solve this:
  - $B_{cl}(i, FCT_n) \leq \tau$?

FCT_n - Broadcast Algorithm for Root Vertices

- Instead of finding $B_{cl}(i, FCT_n)$, solve this:
  - $B_{cl}(i, FCT_n) \leq \tau$?
- **Lemma:**
  - $\underbrace{\max\left\{\lceil \log n \rceil, \max\{B_{cl}(i, T_i)\}\right\}}_{lb} \leq B_{cl}(i, FCT_n) \leq \underbrace{\lceil \log n \rceil + \max\{B_{cl}(i, T_i)\}}_{ub}$

FCT_n - Broadcast Algorithm for Root Vertices

- Instead of finding $B_{cl}(i, FCT_n)$, solve this:
  - $B_{cl}(i, FCT_n) \leq \tau$?
- **Lemma:**
  - $\underbrace{\max\left\{\lceil \log n \rceil, \max\{B_{cl}(i, T_i)\}\right\}}_{lb} \leq B_{cl}(i, FCT_n) \leq \underbrace{\lceil \log n \rceil + \max\{B_{cl}(i, T_i)\}}_{ub}$
- Do a binary search on this range.
  - Invoke the main algorithm (BR_τ) within this method.

FCT_n - Broadcast Algorithm for Root Vertices

- Instead of finding $B_{cl}(i, FCT_n)$, solve this:
  - $B_{cl}(i, FCT_n) \leq \tau$?
- **Lemma:**
  - $\underbrace{\max\left\{\lceil \log n \rceil, \max\{B_{cl}(i, T_i)\}\right\}}_{lb} \leq B_{cl}(i, FCT_n) \leq \underbrace{\lceil \log n \rceil + \max\{B_{cl}(i, T_i)\}}_{ub}$
- Do a binary search on this range.
  - Invoke the main algorithm (BR_τ) within this method.
- Proof of correctness.
- Complexity: $O(|V| \log \log n)$

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Hypercube of Trees

- A Hypercube of Trees HT_k:
  - ◇ A hypercube of dimension k +
  - ◇ 2^k arbitrary trees.
- Current upper bound: An approximation algorithm with a $(2-\varepsilon)$-approximation ratio [2]

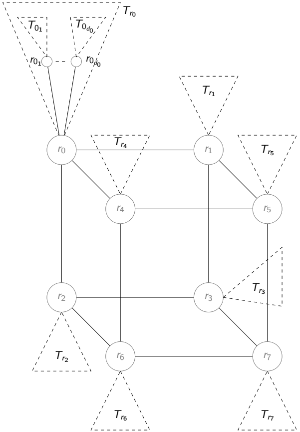


Figure: HT_3, A hypercube of trees with dimension 3

[2] Bhabak, P., Harutyunyan, H. A. (2014). Broadcast problem in hypercube of trees. In International Workshop on Frontiers in Algorithmics (pp. 1–12).

HT_k - Proposed Heuristic

- Instead of finding $B_{cl}(u, HT_k)$, solve this:
  - $B_{cl}(u, HT_k) \leq \tau$?

HT_k - Proposed Heuristic

- Instead of finding $B_{cl}(u, HT_k)$, solve this:
  - $B_{cl}(u, HT_k) \leq \tau$?
- Already know the upper bound and lower bound:
  - $\underbrace{\max\left\{k, \max_{0 \leq i \leq 2^k - 1}\{B_{cl}(r_i, T_i)\}\right\}}_{lb} \leq B_{cl}(u, HT_k) \leq \underbrace{k + \max_{0 \leq i \leq 2^k - 1}\{B_{cl}(r_i, T_i)\}}_{ub}$

HT_k - Proposed Heuristic

- Instead of finding $B_{cl}(u, HT_k)$, solve this:
  - ◇ $B_{cl}(u, HT_k) \leq \tau$?
- Already know the upper bound and lower bound:
  - ◇ $\underbrace{\max\left\{k, \max_{0 \leq i \leq 2^k-1}\{B_{cl}(r_i, T_i)\}\right\}}_{lb} \leq B_{cl}(u, HT_k) \leq \underbrace{k + \max_{0 \leq i \leq 2^k-1}\{B_{cl}(r_i, T_i)\}}_{ub}$
- Do a binary search on this range.
  - ◇ Invoke the main heuristic (BR_τ) within this method.

HT_k - Proposed Heuristic

- Instead of finding $B_{cl}(u, HT_k)$, solve this:
  - $B_{cl}(u, HT_k) \leq \tau$?
- Already know the upper bound and lower bound:
  - $\underbrace{\max\left\{k, \max_{0 \leq i \leq 2^k-1} \{B_{cl}(r_i, T_i)\}\right\}}_{lb} \leq B_{cl}(u, HT_k) \leq \underbrace{k + \max_{0 \leq i \leq 2^k-1} \{B_{cl}(r_i, T_i)\}}_{ub}$
- Do a binary search on this range.
  - Invoke the main heuristic (BR_τ) within this method.
- Our numerical results on graphs of up to 5 million vertices indicate that the heuristic is able to outperform the best-known algorithm for the same problem in up to 90% of the experiments while speeding up the process up to 30%.

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Fully-adaptive Model

- Another sub-model for universal lists,
- A universal list L_v is maintained at each vertex v,
- Once informed, follow the list and **skip all informed vertices!**
  - ◇ Similarly to the classical model: No unnecessary calls!

Fully-adaptive Model

- Another sub-model for universal lists,
- A universal list L_v is maintained at each vertex v,
- Once informed, follow the list and **skip all informed vertices!**
  - Similarly to the classical model: No unnecessary calls!
- **Theorem 3.1.** *$B_{cl}(G) \leq B_{fa}(G) \leq B_a(G) \leq B_{na}(G)$, for any graph G.*

Model	Symbol	No. of unnecessary calls	Required Space	Speed
Non-adaptive	$B_{na}(G)$	Many	$\sum_{1\leq i\leq n} d_i$	Very Slow
Adaptive	$B_a(G)$	Few	$2\times\sum_{1\leq i\leq n} d_i$	Slow
Fully Adaptive	$B_{fa}(G)$	0	$2\times\sum_{1\leq i\leq n} d_i$	Moderate
Classical	$B_{cl}(G)$	0	$n\times\sum_{1\leq i\leq n} d_i$	Very Fast

Fully-adaptive Model - Definitions

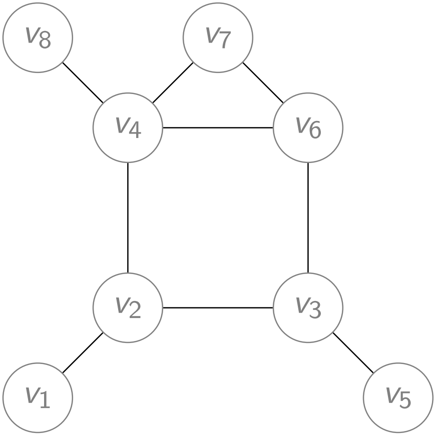
- A broadcast scheme: Matrix $\sigma_{n \times \Delta}$,
  - ◇ Row i of σ corresponds to an ordering of neighbors for vertex v_i.
- Set of all possible schemes: Σ.

Fully-adaptive Model - Definitions

- A broadcast scheme: Matrix $\sigma_{n \times \Delta}$,
  - ◇ Row i of σ corresponds to an ordering of neighbors for vertex v_i.
- Set of all possible schemes: Σ.
- Let $M \in \{na, a, fa\}$ be a model:
  - ◇ $B_M^\sigma(v, G)$: the time steps needed to inform all the vertices in G from v while following σ under M,
  - ◇ $B_M^\sigma(G) = \max_{v \in V}\{B_M^\sigma(v, G)\}$,
  - ◇ $B_M(G) = \min_{\sigma \in \Sigma}\{B_M^\sigma(G)\}$.

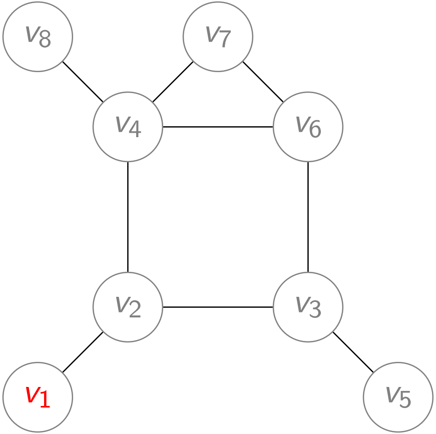
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



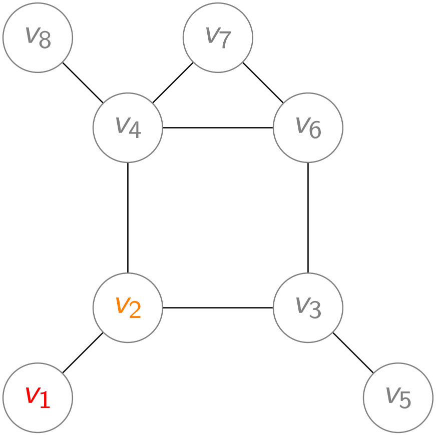
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



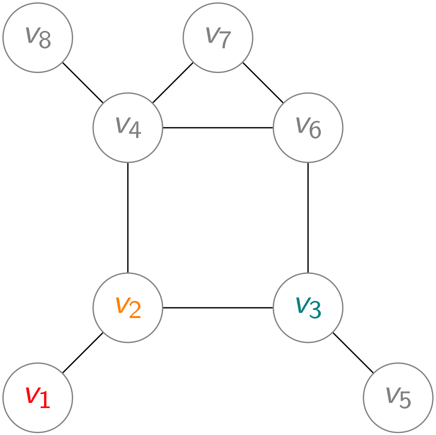
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



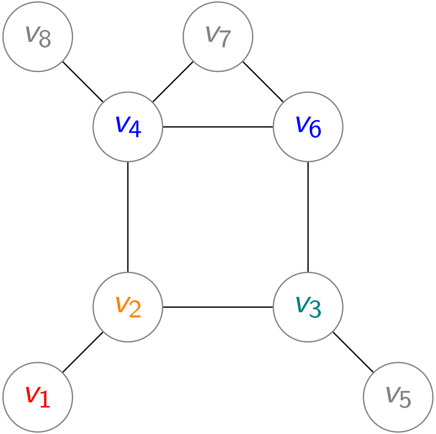
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



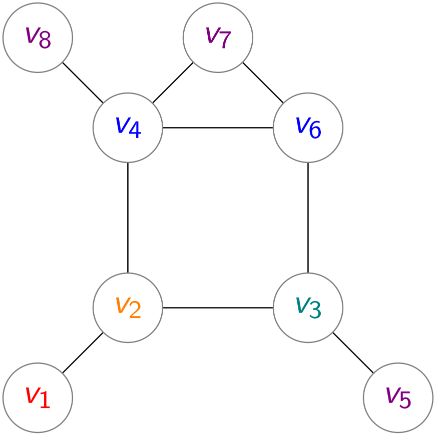
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



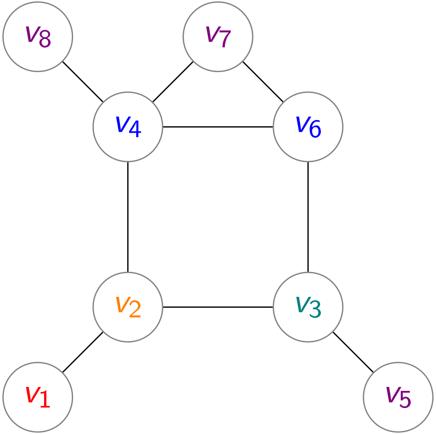
Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



Fully-adaptive model - Example

Sender	Ordering of receivers			
v_1	v_2	Null	Null	Null
v_2	v_3	v_4	v_1	Null
v_3	v_2	v_6	v_5	Null
v_4	v_2	v_6	v_8	v_7
v_5	v_3	Null	Null	Null
v_6	v_3	v_7	v_4	Null
v_7	v_6	v_4	Null	Null
v_8	v_4	Null	Null	Null



- $B^{\sigma}_{fa}(v_1, G) = 4$, while $B^{\sigma}_{a}(v_1, G) = 5$ and $B^{\sigma}_{na}(v_1, G) = 6$.

Fully-adaptive Model - AAA

- **Assumptions:**
  - None-faulty network with established links,
  - Unique and heavy message,
  - The message: header + payload,

Fully-adaptive Model - AAA

- **Assumptions:**
  - None-faulty network with established links,
  - Unique and heavy message,
  - The message: header + payload,
- **Architecture:**
  - How to know the state of each neighbour?
    - Push model,
    - Pull model,

Fully-adaptive Model - AAA

- **Assumptions:**
  - None-faulty network with established links,
  - Unique and heavy message,
  - The message: header + payload,
- **Architecture:**
  - How to know the state of each neighbour?
    - Push model,
    - Pull model,
- **Applications:**
  - Update procedure in SDNs:
    - Changing routing policies, adjusting links' weights, etc.
    - The data plane only forwards packets,
    - Routing and load balancing decisions are made in a centralized controller,
    - The network manager must optimize the forwarding tables (broadcast schemes).

Results on fully-adaptive model

- Trees T:
  - **Theorem 5.2.1.** $B_{cl}(T) = B_{fa}(T) = B_{a}(T)$.

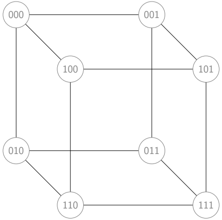
Results on fully-adaptive model

- Trees T:
  - ◇ **Theorem 5.2.1.** $B_{cl}(T) = B_{fa}(T) = B_a(T)$.
- Grids $G_{m \times n}$:
  - ◇ **Corollary 5.2.1.** $B_{fa}(G_{m \times n}) = m + n - 2$.

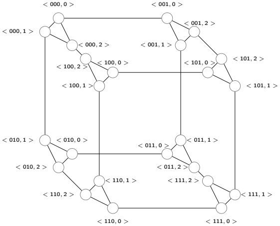
Results on fully-adaptive model

- Trees T:
  - **Theorem 5.2.1.** $B_{cl}(T) = B_{fa}(T) = B_a(T)$.
- Grids $G_{m\times n}$:
  - **Corollary 5.2.1.** $B_{fa}(G_{m\times n}) = m + n - 2$.
- Tori $T_{m\times n}$:
  - **Theorem 5.2.2.**
    - $B_{fa}(T_{m\times n}) = \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor$, if n and m are even,
    - $B_{fa}(T_{m\times n}) = \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1$, if one of m and n is even and the other one is odd,
    - $\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1 \leq B_{fa}(T_{m\times n}) \leq \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 2$, if both m and n are odd.

Results on fully-adaptive model - cont.



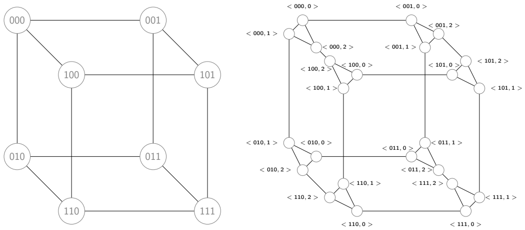
a) H_3



b) CCC_3

- Hypercubes H_d:
  - ◇ **Theorem 7.1.2.** $B_{fa}(H_d) = d$.

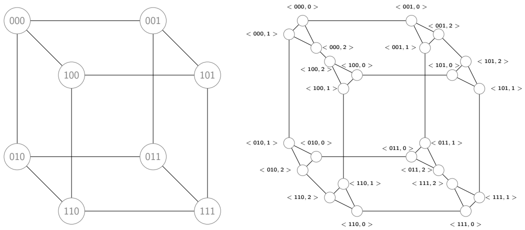
Results on fully-adaptive model - cont.



a) H_3 *b)* CCC_3

- Hypercubes H_d:
  - **Theorem 7.1.2.** $B_{fa}(H_d) = d$.
    - **Corollary 7.1.4.** *Hypercube H_d is an mbg on 2^d vertices under the fully-adaptive model.*

Results on fully-adaptive model - cont.



a) H_3 *b)* CCC_3

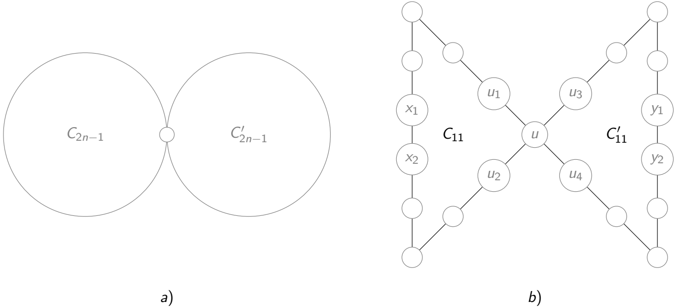
- Hypercubes H_d:
  - ◇ **Theorem 7.1.2.** $B_{fa}(H_d) = d$.
    - ◇ **Corollary 7.1.4.** *Hypercube* H_d *is an mbg on* 2^d *vertices under the fully-adaptive model.*
- Cube Connected Cycles CCC_d:
  - ◇ **Theorem 5.2.3.** $B_{fa}(CCC_d) = \lceil \frac{5d}{2} \rceil - 1$.

Results on fully-adaptive model - cont.

- Is $B_{cl}(G) = B_{fa}(G)$ always?

Results on fully-adaptive model - cont.

- Is $B_{cl}(G) = B_{fa}(G)$ always?
  - ◇ No!
  - ◇ **Proposition 5.2.1.** *There exists graph G with $B_{cl}(G) < B_{fa}(G)$:*



◇

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - ◇ **Theorem 6.1.1.** $B_{na}(T_{k,h}) = kh + 2h - 1.$

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - ◇ **Theorem 6.1.1.** $B_{na}(T_{k,h}) = kh + 2h - 1$.
- Binomial trees T_d:
  - ◇ **Proposition 6.1.1.** $B_{na}(T_d) = 3d - 2$.

Results on non-adaptive model

- Complete $k-$ary trees $T_{k,h}$:
  - **Theorem 6.1.1.** $B_{na}(T_{k,h}) = kh + 2h - 1$.
- Binomial trees T_d:
  - **Proposition 6.1.1.** $B_{na}(T_d) = 3d - 2$.
- Complete Bipartite graph $K_{m\times n}$:
  - **Theorem 6.1.2.** $B_{cl}(K_{m\times n}) = \lceil \log n \rceil + 1 + \max\{\lceil \frac{m-2^{\lceil \log n \rceil}}{n} \rceil, 0\}$.
  - **Theorem 6.1.3.** $B_{na}(K_{m\times n}) \leq B_{cl}(K_{m\times n}) + 3 \times \lceil \sqrt{B_{cl}(K_{m\times n})} \rceil$.

Results on non-adaptive model - cont.

- A general upper bound for trees:
  - **Theorem 6.1.4.** $B_{na}(T) \leq B_{cl}(T) + \lfloor \frac{diam(T)}{2} \rfloor$.
- Tightest bounds on trees:
  - **Theorem 6.1.5.**

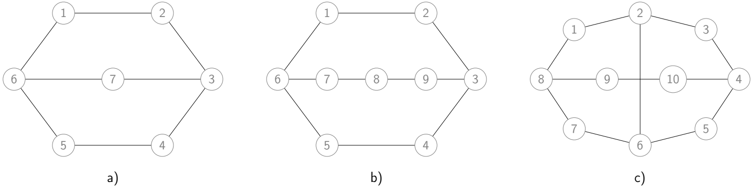
$$\max\left\{B_{cl}(T)+1, \lceil \frac{3.diam(T)-1}{2} \rceil\right\} \leq B_{na}(T) \leq \min\left\{B_{cl}(T) + \lfloor \frac{diam(T)}{2} \rfloor, b_{cl}(T)+diam(T)\right\} \quad (1)$$

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

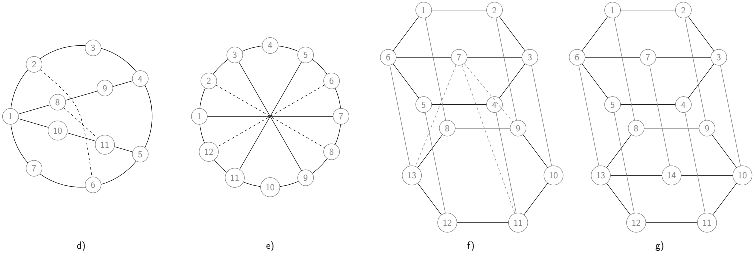
Broadcast graphs under fully-adaptive model

- Graph G on n vertices is a broadcast graph *(bg)* if $B_{fa}(G) = \lceil \log n \rceil$,
- A *bg* with the minimum number of edges is called a minimum broadcast graph *(mbg)*,
- The number of edges of an *mbg* on n vertices: $B^{(fa)}(n)$.
- **Lemma 7.1.1.** *If there is a graph G on n vertices for which $B_{fa}(G) = \lceil \log n \rceil$, then $B^{(cl)}(n) \leq B^{(fa)}(n)$.*
- mbg's for $n \leq 10$:



Broadcast graphs under fully-adaptive model - cont.

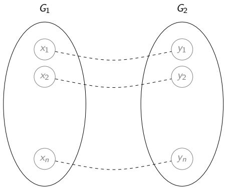
- bg's for $11 \leq n \leq 14$:



n	3	4	5	6	7	8	9	10	11	12	13	14
Lower bound on $B^{(fa)}(n)$	2	4	5	6	8	12	10	12	13	15	18	21
Upper bound on $B^{(fa)}(n)$	2	4	5	6	8	12	10	12	15	17	23	23

Broadcast graphs under fully-adaptive model - cont.

- General construction of bg's:
  - ◇ **Lemma 7.1.2.** *Consider a graph* $G = (V, E)$ *with* n *vertices,* m *edges, and* $B_{fa}(G) = \tau$*. It is always possible to construct a graph* $G' = (V', E')$ *with* $2n$ *vertices,* $2m + n$ *edges, and* $B_{fa}(G') = \tau + 1$*.*



Broadcast graphs under fully-adaptive model - cont.

- This yields 4 infinite families of bg's under fully-adaptive model:
  - ◇ **Theorem 7.1.1.** *For any integer* $k = \lceil \log n \rceil \geq 4$*:*
    $B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-4}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{8n}{9}$,
    $B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-3}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{4n}{5}$,
    $B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-2}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{n}{2}$,
    $B^{(fa)}(n) = B^{(fa)}(2^{k-1} + 2^{k-2} + 2^{k-3}) \leq \frac{n\lceil \log n \rceil}{2} - \frac{5n}{14}$.

Comparing broadcast time of various graphs

Graph G	$B_{cl}(G)$	$B_{fa}(G)$	$B_a(G)$	$B_{na}(G)$
Path P_n	$n-1$	$n-1$	$n-1$	$\lceil \frac{3n}{2} \rceil - 2$
Cycle C_n	$\lceil \frac{n}{2} \rceil$,	$\lceil \frac{n}{2} \rceil$	$\lceil \frac{n}{2} \rceil$	$\lfloor \frac{2n}{3} \rfloor$
Star S_n	$n-1$	$n-1$	$n-1$	n
Complete graph K_n	$\lceil \log n \rceil$	$\leq \lceil \log n \rceil + 2\lceil \sqrt{\log n} \rceil$	$\leq \lceil \log n \rceil + 2\lceil \sqrt{\log n} \rceil$	$\leq \lceil \log n \rceil + 2\lceil \sqrt{\log n} \rceil$
Complete Bipartite $K_{m\times n}$	$t_1 = \lceil \log n \rceil + 1 + \max\{\lceil \frac{m-2^{\lceil \log n \rceil}}{n} \rceil, 0\}$	$\leq t_1 + 3\lceil \sqrt{t_1} \rceil$	$\leq t_1 + 3\lceil \sqrt{t_1} \rceil$	$\leq t_1 + 3\lceil \sqrt{t_1} \rceil$
Complete $k-$ary tree $T_{k,h}$	$kh + h - 1$	$kh + h - 1$	$kh + h - 1$	$kh + 2h - 1$
Binomial tree T_d	$2d - 1$	$2d - 1$	$2d - 1$	$3d - 2$
Grid $G_{m\times n}$	$m + n - 2$	$m + n - 2$	$m + n - 2$	$m + n - 1$
Tori $T_{m\times n}$	$\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor$, if m and n are even $\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1$, otherwise	$\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor$, if m and n are even $\lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 1$, if only one of m and n is even $\leq \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 2$, otherwise	$\leq \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 3$	$\leq \lfloor \frac{n}{2} \rfloor + \lfloor \frac{m}{2} \rfloor + 5$
Hypercube H_d	d	d	$\leq \frac{d(d-1)}{2} + 1$	$\leq \frac{d(d+1)}{2} + 1$
Cube Connected Cycle CCC_d	$\lceil \frac{5d}{2} \rceil - 1$	$\lceil \frac{5d}{2} \rceil - 1$	$\leq 2\lceil \frac{5d}{2} \rceil - 1$	$\leq 3\lceil \frac{5d}{2} \rceil - 3$
Shuffle Exchange SE_d	$2d - 1$	$\leq 4d - 1$	$\leq 4d - 1$	$\leq 6d - 3$
De Bruijn DB_d	$\leq \frac{3}{2}(d+1)$, $\geq 1.3171d$	$\leq 3d + 1$	$\leq 3d + 1$	$\leq 4d$

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Introduction

- Proposing the first heuristic for broadcasting with universal lists:
- Given a graph G and a model $M \in \{fa, a, na\}$, find a broadcast scheme $\sigma \in \Sigma$ that minimizes $B_M^\sigma(G)$.
- Why this problem is difficult?

Introduction

- Proposing the first heuristic for broadcasting with universal lists:
- Given a graph G and a model $M \in \{fa, a, na\}$, find a broadcast scheme $\sigma \in \Sigma$ that minimizes $B_M^\sigma(G)$.
- Why this problem is difficult?
  - **Proposition 8.2.1.** For a graph G on n vertices, where the degree of vertex i is d_i, the size of search space for the problem of broadcasting using universal list is as follows:

$$|\Sigma_{(G)}| = \prod_{i=1}^{n} \sum_{j=0}^{d_i} \left(\binom{d_i}{j} \times j! \right) \qquad (2)$$

  - Exponential growth!

HUB-GA: Methodology

- HUB-GA: a Heuristic for Universal list model of Broadcasting with Genetic Algorithm.
- GA: a population based search algorithm.

HUB-GA: Methodology

- HUB-GA: a Heuristic for Universal list model of Broadcasting with Genetic Algorithm.
- GA: a population based search algorithm.
  - Each solution to the problem is a Chromosome,
  - The fitness of each individual is evaluated with a fitness function.
  - To improve the quality of solutions, the best solutions are selected for reproduction using two primary operations of GA: Crossover and Mutation.
  - GA tries to find a suitable solution by repeating this process over multiple generations.

HUB-GA: Methodology

Algorithm 6 HUB-GA

```
Generate random population;
Calculate fitness score;
while not converged do
    Crossover;
    Mutation;
    Calculate fitness score;
    Acceptance;
end while
return The best chromosome
```

HUB-GA: Genes, Chromosomes, and Population

- Consider a graph G with n vertices, where d_i = the degree of vertex i.

HUB-GA: Genes, Chromosomes, and Population

- Consider a graph G with n vertices, where d_i = the degree of vertex i.
- *Gene* i, $g^{(i)}$: An arbitrary ordering of the neighbors of vertex i with size at most d_i.

HUB-GA: Genes, Chromosomes, and Population

- Consider a graph G with n vertices, where d_i = the degree of vertex i.
- *Gene* i, $g^{(i)}$: An arbitrary ordering of the neighbors of vertex i with size at most d_i.
- *Chromosome* is a collection of n genes: $g^{(1)}, g^{(2)}, \cdots, g^{(n)}$.
  - ◇ A chromosome is a matrix σ with n rows (or n genes) and Δ columns.
  - ◇ In GA, a chromosome is a possible solution for the problem: any $\sigma \in \Sigma$ may be an optimal broadcast scheme.

1	2	3		
2	1	4	3	
3	2	4	1	5
4	3	2	6	
5	6	3		
6	5	4		

HUB-GA: Genes, Chromosomes, and Population

- Consider a graph G with n vertices, where d_i = the degree of vertex i.
- *Gene* i, $g^{(i)}$: An arbitrary ordering of the neighbors of vertex i with size at most d_i.
- *Chromosome* is a collection of n genes: $g^{(1)}, g^{(2)}, \cdots, g^{(n)}$.
  - ◇ A chromosome is a matrix σ with n rows (or n genes) and Δ columns.
  - ◇ In GA, a chromosome is a possible solution for the problem: any $\sigma \in \Sigma$ may be an optimal broadcast scheme.
  - ◇ Guessing the optimal chromosome out of many possible solutions is impossible.

1	2	3		
2	1	4	3	
3	2	4	1	5
4	3	2	6	
5	6	3		
6	5	4		

HUB-GA: Genes, Chromosomes, and Population

- Consider a graph G with n vertices, where d_i = the degree of vertex i.
- *Gene* i, $g^{(i)}$: An arbitrary ordering of the neighbors of vertex i with size at most d_i.
- *Chromosome* is a collection of n genes: $g^{(1)}, g^{(2)}, \cdots, g^{(n)}$.
  - ◇ A chromosome is a matrix σ with n rows (or n genes) and Δ columns.
  - ◇ In GA, a chromosome is a possible solution for the problem: any $\sigma \in \Sigma$ may be an optimal broadcast scheme.
  - ◇ Guessing the optimal chromosome out of many possible solutions is impossible.
- The first step of HUB-GA: generate $|p|$ solutions randomly, called the first *population*.

1	2	3		
2	1	4	3	
3	2	4	1	5
4	3	2	6	
5	6	3		
6	5	4		

HUB-GA: Fitness function

- The fitness function, $f(\sigma)$, evaluates the fitness of a chromosome σ.
- $f(\sigma)$ should be minimized when σ is an optimal solution.

HUB-GA: Fitness function

- The fitness function, $f(\sigma)$, evaluates the fitness of a chromosome σ.
- $f(\sigma)$ should be minimized when σ is an optimal solution.
  - ◇ $f_1(\sigma)$: the broadcast time:

$$f_1(\sigma) = \max_{u \in V(G)} \{B_M^{\sigma}(u, G)\} = B_M^{\sigma}(G) \tag{3}$$

  - ◇ $f_2(\sigma)$: average broadcast time:

$$f_2(\sigma) = \frac{\sum_{u \in V(G)} B_M^{\sigma}(u, G)}{n} \tag{4}$$

HUB-GA: Crossover

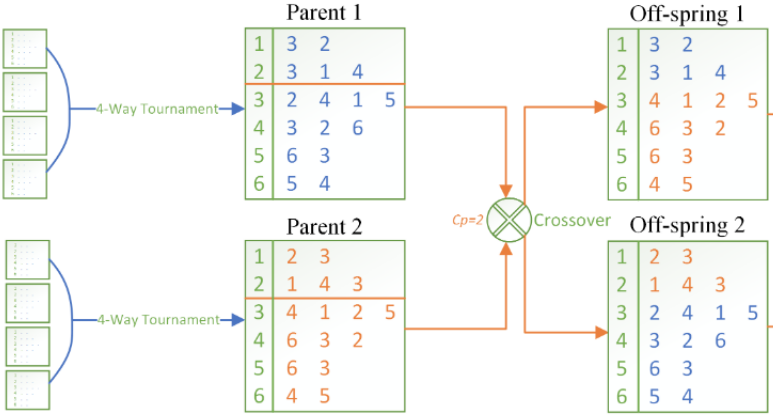
- Two chromosomes are selected as the parents (*selection* phase), and then two children (called *offsprings*) are generated by *crossover*.

HUB-GA: Crossover

- Two chromosomes are selected as the parents (*selection* phase), and then two children (called *offsprings*) are generated by *crossover*.
- *Selection*: $K-$way tournament: select the fittest chromosome among K randomly chosen chromosomes.

HUB-GA: Crossover

- Two chromosomes are selected as the parents (*selection* phase), and then two children (called *offsprings*) are generated by *crossover*.
- *Selection*: $K-$way tournament: select the fittest chromosome among K randomly chosen chromosomes.

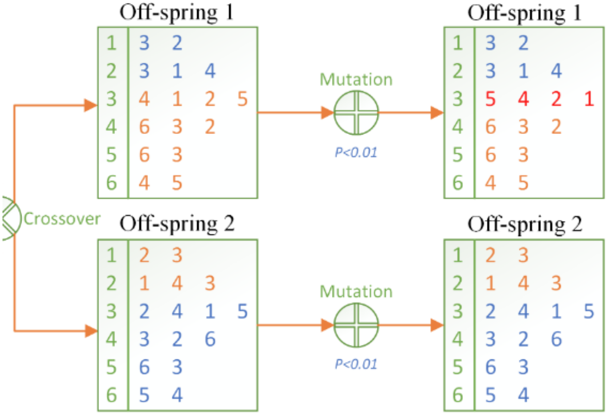


HUB-GA: Mutation

- Mutation: A gene of an offspring is changed randomly with a small probability.
- In our algorithm: shuffle the ordering of a randomly selected gene.

HUB-GA: Mutation

- Mutation: A gene of an offspring is changed randomly with a small probability.
- In our algorithm: shuffle the ordering of a randomly selected gene.



HUB-GA: Acceptance

- After doing Crossover and Mutation, the population size grows.
- One possible solution to keep the current generation manageable with limited resources is to retain the original population size by allowing a fixed number of chromosomes to survive into the next generation.

HUB-GA: Acceptance

- After doing Crossover and Mutation, the population size grows.
- One possible solution to keep the current generation manageable with limited resources is to retain the original population size by allowing a fixed number of chromosomes to survive into the next generation.
  - $K-$way tournament.

HUB-GA: Stopping criterion

- The execution of HUB-GA terminates if, after S_t iterations, the fitness score of the fittest individual does not change drastically (*convergence*).
- Once the stopping criterion is met, the best chromosome (solution) from the current generation and its fitness score are returned as the final answer.

HUB-GA: Remarks

- The first heuristic for this problem,
- Working for arbitrary graphs,
- Working for any model under universal lists
- Possibility of defining various fitness scores,
- Providing the broadcast scheme.

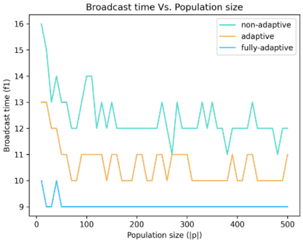
HUB-GA: Experimental setup

Experiment	What?	Why?	How?	Graph(s)
Experiment 1	Parameter Tuning	To see the impact of changing HUB−GA parameters on its performance.	For a graph G, we change parameters $\|p\|$ and S_t, while reporting $f_1(\sigma)$ and $f_2(\sigma)$ and the run time.	Karate club network (Zachary, 1977)
Experiment 2	Performance comparison vs. Classical model	To see whether the found broadcast time under universal lists model approaches its optimal value or not.	By calculating the ratio of $B_M(G)/B_{cl}(G)$ for different interconnection networks.	Well-known interconnection networks for which the value of $B_{cl}(G)$ is known.
Experiment 3	Performance comparison vs. degree-based heuristics	To see whether HUB−GA outperforms other degree-based heuristics or not.	By comparing the performance of our heuristic with three heuristics for clique-like structure graphs.	Clique-like graphs: Ring of cliques (Kamiński, Prałat, & Théberge, 2021), and Windmill graph (Bermond, 1979)
Experiment 4	Performance comparison vs. state-of-the-art heuristics	To see whether HUB−GA gets close to other heuristics for classical broadcasting or not.	By comparing the performance of our heuristic with two lower bounds and six upper bounds.	Interconnection Networks and Complex networks with small-world model (Rossi & Ahmed, 2015)

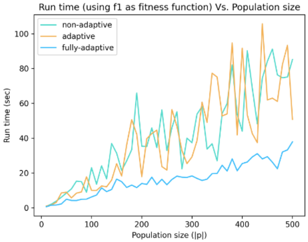
HUB-GA: Experiment 1

- Change $|p|$ and S_t, report $f_1(\sigma)$ and $f_2(\sigma)$ and the run time.
- Choosing $|p|$ is a trade-off. The bigger the $|p|$:
  - The higher the chance of finding a near-optimal solution in early iterations.
  - The higher the computational cost.
- The same is true for S_t

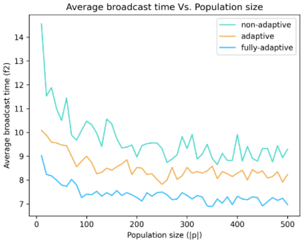
HUB-GA: Experiment 1, $|p|$



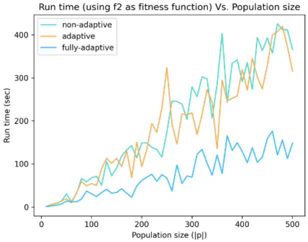
(a) Broadcast time (f_1)



(b) Run time for calculating f_1

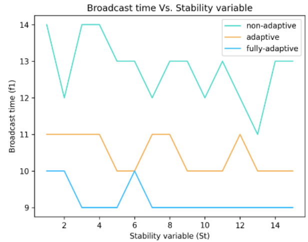


(c) Average broadcast time (f_2)

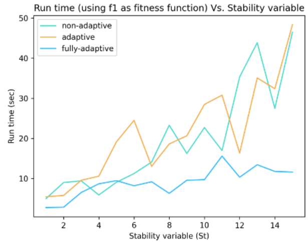


(d) Run time for calculating f_2

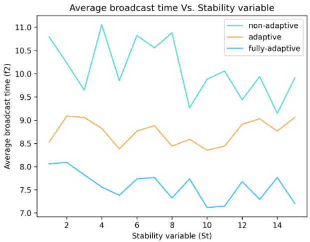
HUB-GA: Experiment 1, S_t



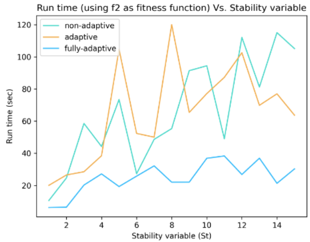
(a) Broadcast time (f_1)



(b) Run time for calculating f_1



(c) Average broadcast time (f_2)



(d) Run time for calculating f_2

HUB-GA: Experiment 2

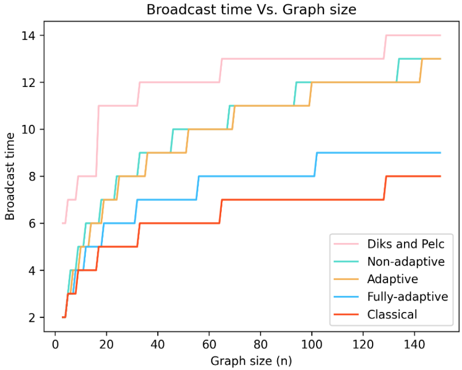
- Compare the GA heuristic for the universal list model with known bounds on the classical model for commonly used interconnection networks.

HUB-GA: Experiment 2

Graph G	n	$\frac{B^{\sigma}_{fa}(G)}{B_{cl}(G)}$	$\frac{B^{\sigma}_{a}(G)}{B_{cl}(G)}$	$\frac{B^{\sigma}_{na}(G)}{B_{cl}(G)}$
Path P_n	$2 \leq n \leq 1000$	**1.00***	**1.00***	**1.49***
Cycle C_n	$3 \leq n \leq 1000$	**1.00***	**1.00***	**1.32***
Star S_n	$2 \leq n \leq 1000$	**1.00***	**1.00***	**1.01***
Complete Graph K_n	$3 \leq n \leq 50$	1.14	1.39	1.42
Grid $G_{n \times m}$	$2 \leq n, m \leq 10$	1.07	1.08	1.35
Tori $T_{n \times m}$	$2 \leq n, m \leq 10$	1.09	1.24	1.55
Hypercube H_d	$2 \leq d \leq 9$	1.06	**1.41***	**1.68***
Cube Connected Cycle CCC_d	$2 \leq d \leq 7$	1.14	**1.18***	**1.52***
Shuffle Exchange SE_d	$3 \leq d \leq 9$	**1.06***	**1.09***	**1.44***
De Bruijn DB_d	$3 \leq d \leq 9$	**1.09***	**1.18***	**1.51***

- **Conjecture 8.4.1.** For a sufficiently large n, the broadcast time of a complete graph K_n is bounded as follows:

$$\lceil \log n \rceil = B_{cl}(K_n) < B_{fa}(K_n) < B_a(K_n) \leq B_{na}(K_n) \leq \lceil \log n \rceil + 2\lceil \sqrt{\log n} \rceil \quad (5)$$



- Compare the GA heuristic with degree-based heuristics:
  - ◇ `Ran. Seq.`: The ordering of a vertex is uniformly random.
  - ◇ `Inc. Deg.`: Neighbors of a vertex are sorted in ascending order based on their degree.
  - ◇ `Dec. Deg.`: Neighbors of a vertex are sorted in descending order based on their degree.
- For graphs with clique-like subgraphs:

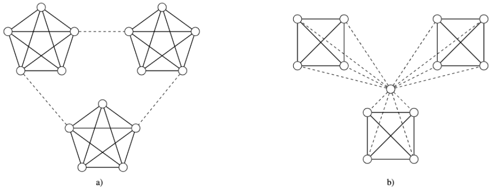


Figure 8.8: a) Ring of Clique $RC_{3,5}$, b) Windmill graph $W_{5,3}$

HUB-GA: Experiment 3, $RC_{n,m}$

$RC_{n,m}$		$\|V\|$	$\|E\|$	Non-adaptive model											
				Min				Avg				Max			
n	m			Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA
3	3	9	12	**5**	**5**	6	**5**	6.22	6.11	6.77	**5.33**	7.4	7	7	**6**
4	3	12	16	6.6	7	9	**6**	8.38	7.91	9.41	**6.83**	10.4	9	10	**8**
5	3	15	20	8.4	8	10	**7**	9.93	9	10.86	**8.33**	12	**10**	12	**10**
6	3	18	24	10.2	**9**	12	**9**	12.04	10.44	13.77	**9.66**	13.8	12	15	**10**
3	4	12	21	6.2	6	9	**5**	8.06	7.08	9.5	**5.83**	9.6	8	10	**7**
4	4	16	28	8.4	**7**	12	**7**	10.33	8.81	13.56	**7.81**	12.6	10	15	**9**
5	4	20	35	10.8	**8**	15	**8**	13.05	10.2	16.1	**9.45**	15.8	**12**	17	**12**
6	4	24	42	12.2	**10**	18	**10**	14.91	11.79	19.95	**11.29**	17.8	14	22	**13**
3	5	15	33	7.2	7	12	**6**	9.78	7.93	12.4	**7.2**	11.8	**9**	14	**9**
4	5	20	44	8.8	8	16	**7**	12.37	9.9	17.75	**8.65**	15.4	**11**	19	**11**
5	5	25	55	11.6	9	21	**8**	14.55	11.12	21.84	**10.04**	17.6	13	23	**12**
6	5	30	66	14	**11**	24	**11**	17.91	12.73	26.86	**12.13**	21.8	**14**	29	**14**
3	6	18	48	8.8	8	15	**6**	10.94	8.77	15.77	**7.55**	13	10	17	**9**
4	6	24	64	11	9	20	**7**	14.52	10.29	21.83	**9.2**	17.8	12	24	**11**
5	6	30	80	14	11	25	**9**	17.22	12	26.53	**10.8**	20	**13**	27	**13**
6	6	36	96	16.4	**11**	30	**11**	20.59	13.66	33.05	**12.44**	23.8	16	36	**14**

HUB-GA: Experiment 3, $RC_{n,m}$

$RC_{n,m}$		$\|V\|$	$\|E\|$	Adaptive model											
				Min				Avg				Max			
n	m			Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA
3	3	9	12	4.4	**4**	5	**4**	4.8	**4.33**	5	**4.33**	5	**5**	5	**5**
4	3	12	16	5.6	**5**	6	**5**	6.4	**5.33**	6.66	**5.33**	7	**6**	7	**6**
5	3	15	20	6.8	**6**	8	**6**	7.58	**6.33**	8	**6.33**	8	**7**	8	**7**
6	3	18	24	8	**7**	9	**7**	8.86	**7.33**	9.38	7.61	10	**8**	10	**8**
3	4	12	21	5.6	5	6	**4**	6.13	5.41	6.83	**5.08**	6.8	**6**	7	**6**
4	4	16	28	7	6	9	**5**	8.36	6.5	9.12	**6.25**	9.4	**7**	10	**7**
5	4	20	35	8.4	7	11	**6**	9.77	**7.25**	11	**7.25**	10.8	**8**	11	**8**
6	4	24	42	10.4	**8**	12	**8**	12.15	**8.29**	12.91	8.41	13.6	**9**	13	**9**
3	5	15	33	6.6	**5**	8	**5**	7.71	6	8.13	**5.6**	8.6	7	9	**6**
4	5	20	44	7.8	**6**	11	**6**	9.45	7.35	11.35	**6.8**	11	**8**	12	**8**
5	5	25	55	9.8	**7**	12	**7**	11.52	**7.64**	12.4	7.96	13.2	**8**	13	9
6	5	30	66	12	**8**	14	**8**	13.97	9.16	15.26	**9.03**	15.8	**10**	17	**10**
3	6	18	48	6.8	**6**	9	**6**	8.46	7.05	9.55	**6.44**	9.6	8	10	**7**
4	6	24	64	9	7	13	**6**	11.12	7.69	13.04	**7.45**	13	**8**	14	**8**
5	6	30	80	11	**8**	14	**8**	13.15	8.93	14.5	**8.86**	14.8	**10**	15	**10**
6	6	36	96	13.8	9	17	**8**	15.66	9.83	18	**9.8**	17.6	**11**	19	**11**

HUB-GA: Experiment 3, $RC_{n,m}$

$RC_{n,m}$		$\|V\|$	$\|E\|$	Fully-adaptive model											
				Min				Avg				Max			
n	m			Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA	Ran.Seq.	Dec.Deg.	Inc.Deg.	HUB-GA
3	3	9	12	4.4	**4**	5	**4**	4.86	**4.33**	5	**4.33**	5	**5**	5	**5**
4	3	12	16	5.8	**5**	6	**5**	6.25	**5.33**	6.41	**5.33**	7	**6**	7	**6**
5	3	15	20	6.6	**6**	8	**6**	7.53	**6.33**	8	**6.33**	8	**7**	8	**7**
6	3	18	24	8	**7**	9	**7**	8.73	**7.33**	9.44	**7.33**	9.2	**8**	10	**8**
3	4	12	21	5.4	**4**	6	5	5.9	**5**	6.66	**5**	6.6	6	7	**5**
4	4	16	28	6.6	**5**	8	**5**	7.82	5.87	8.81	**5.5**	9	7	10	**6**
5	4	20	35	8.2	**6**	11	**6**	9.36	7.1	11	**7.05**	10.4	**8**	11	**8**
6	4	24	42	10.2	**7**	13	**7**	11.23	8.25	13.29	**8.16**	13	**9**	14	**9**
3	5	15	33	6	**5**	7	**5**	6.7	**5.6**	7.86	5.73	7.2	**6**	9	**6**
4	5	20	44	7.6	**6**	9	**6**	9.05	6.66	9.9	**6.65**	10.2	**7**	11	**7**
5	5	25	55	9.4	**7**	12	**7**	10.7	7.92	13.04	**7.8**	12.2	**9**	14	**9**
6	5	30	66	11	**8**	14	**8**	12.54	**8.6**	15.03	9.16	14	**9**	16	10
3	6	18	48	6.8	6	8	**5**	7.76	6.66	9.27	**5.88**	8.6	**7**	10	**7**
4	6	24	64	8.6	**7**	12	**7**	10.17	7.95	12.83	**7.5**	11.6	9	14	**8**
5	6	30	80	10.4	**7**	13	8	11.66	**8.26**	14.03	9.1	13	**9**	15	10
6	6	36	96	11.8	**8**	16	9	13.71	**9.25**	17.16	10.27	15.4	**10**	19	11

HUB-GA: Experiment 4

- Compare the GA heuristic with state-of-the-art heuristics for classical broadcasting:
  - Two lower bounds on $B_{cl}(v, G)$: `TLB`, `LBB`,
  - Six upper bounds on $B_{cl}(v, G)$: `TreeBlock`, `NTBA`, `NEWH`, `ILP`, `ACS`, `BRKGA`.
- For two types of networks:
  - Interconnection networks (44 instances),
  - Connected complex networks (30 instances).

HUB-GA: Experiment 4, Interconnection networks

Instance	$\lvert V\rvert$	$\lvert E\rvert$	Density	*LB* on $B_{cl}(v,G)$		*UB* on $B_{cl}(v,G)$						HUB-GA		
				TLB	LBB	TreeBlock	NTBA	NEWH	ILP	ACS	BRKGA	$B^{\sigma}_{fa}(G)$	$B^{\sigma}_{a}(G)$	$B^{\sigma}_{na}(G)$
H_2	4	4	0.6667	2	-	-	-	-	-	-	-	2	2	2
H_3	8	12	0.4285	3	-	-	-	-	-	-	-	3	4	4
H_4	16	32	0.2667	4	-	-	-	-	4	-	-	4	5	6
H_5	32	80	0.1613	5	5	5	5	5	5	5	5	5	7	9
H_6	64	192	0.0952	6	6	6	7	6	6	6	6	7	9	11
H_7	128	448	0.0551	7	7	7	9	7	7	7	7	8	11	14
H_8	256	1024	0.0314	8	8	8	11	8	8	8	9	9	13	16
H_9	512	2304	0.0176	9	9	9	14	9	9	9	10	10	15	18
CCC_2	8	12	0.4285	3	-	-	-	-	-	-	-	4	4	5
CCC_3	24	36	0.1304	5	6	-	6	7	6	6	6	8	8	10
CCC_4	64	94	0.0476	6	8	-	7	9	9	9	9	11	11	15
CCC_5	160	240	0.0189	8	10	-	11	12	11	12	11	14	15	19
CCC_6	384	576	0.0078	9	13	-	14	14	13	14	13	17	18	24
DB_3	8	16	0.5714	3	-	-	4	4	-	-	-	4	4	5
DB_4	16	32	0.2583	4	4	4	5	5	-	5	5	6	6	7
DB_5	32	64	0.1270	5	5	7	7	7	-	6	6	7	8	10
DB_6	64	128	0.0630	6	6	8	8	8	-	8	8	9	10	13
DB_7	128	256	0.0314	7	7	12	10	10	-	10	9	11	12	16
DB_8	256	512	0.0157	8	8	12	12	12	-	12	11	13	14	19
DB_9	512	1024	0.0078	9	9	14	13	13	-	14	13	15	17	22
SE_2	4	5	0.8334	2	-	-	-	-	-	-	-	3	3	4
SE_3	8	12	0.4285	3	-	-	5	5	-	-	-	5	5	6
SE_4	16	21	0.1750	4	7	-	7	7	7	7	7	7	7	9
SE_5	32	46	0.0927	5	9	-	9	9	9	9	9	9	10	13
SE_6	64	93	0.0461	6	11	-	11	11	11	11	11	12	12	16
SE_7	128	190	0.0234	7	13	-	13	13	13	13	13	15	15	20
SE_8	256	381	0.0117	8	15	-	15	15	15	15	15	17	17	25
SE_9	512	766	0.0059	9	17	-	18	18	17	17	18	20	21	28
$H_{10,30}$	30	150	0.3448	5	3	6	-	-	5	5	5	7	9	9
$H_{11,50}$	50	275	0.2245	6	3	7	-	-	6	6	6	8	10	11
$H_{20,50}$	50	500	0.4082	6	3	8	-	-	6	6	6	8	10	11
$H_{21,50}$	50	525	0.4286	6	2	7	-	-	6	6	6	7	10	10
$H_{2,100}$	100	100	0.0202	7	50	50	-	-	50	50	50	50	50	67
$H_{2,17}$	17	17	0.1250	4	8	9	-	-	9	9	9	9	9	11
$H_{2,30}$	30	30	0.0690	5	15	15	-	-	15	15	15	15	15	20
$H_{2,50}$	50	50	0.0408	6	25	25	-	-	25	25	25	25	25	29
$H_{3,17}$	17	26	0.1912	4	4	5	-	-	5	5	5	6	6	8
$H_{3,30}$	30	45	0.1034	5	8	9	-	-	9	9	9	9	9	12
$H_{3,50}$	50	75	0.0612	6	13	14	-	-	14	14	14	14	15	17
$H_{5,17}$	17	43	0.3162	4	3	5	-	-	5	5	5	5	6	7
$H_{6,17}$	17	51	0.3750	4	3	5	-	-	5	5	5	6	6	7
$H_{7,17}$	17	60	0.4412	4	2	5	-	-	5	5	5	5	6	6
$H_{8,30}$	30	120	0.2759	5	4	6	-	-	5	6	5	8	9	10
$H_{9,30}$	30	135	0.3103	5	3	6	-	-	5	5	5	7	8	9

HUB-GA: Experiment 4, Connected complex networks

Instance	$\lvert V \rvert$	$\lvert E \rvert$	Density	*LB* on $B_{cl}(v,G)$		*UB* on $B_{cl}(v,G)$						HUB-GA		
				TLB	LBB	TreeBlock	NTBA	NEWH	ILP	ACS	BRKGA	$B^{\sigma}_{fa}(G)$	$B^{\sigma}_{a}(G)$	$B^{\sigma}_{na}(G)$
SW-100-3-0d1-trial1	100	100	0.0202	7	61	-	-	-	61	61	61	68	68	104
SW-100-3-0d2-trial1	100	100	0.0202	7	31	-	-	-	31	31	31	40	40	60
SW-100-3-0d2-trial3	100	100	0.0202	7	31	-	-	-	31	31	31	49	49	74
SW-100-4-0d1-trial1	100	200	0.0404	7	7	-	-	-	9	10	9	14	14	19
SW-100-4-0d1-trial2	100	200	0.0404	7	7	-	-	-	8	9	8	13	14	18
SW-100-4-0d1-trial3	100	200	0.0404	7	9	-	-	-	10	11	10	15	16	20
SW-100-4-0d2-trial1	100	200	0.0404	7	7	-	-	-	8	9	8	12	13	17
SW-100-4-0d2-trial2	100	200	0.0404	7	7	-	-	-	8	9	9	12	13	16
SW-100-4-0d2-trial3	100	200	0.0404	7	7	-	-	-	9	9	9	12	13	17
SW-100-4-0d3-trial1	100	200	0.0404	7	6	-	-	-	8	9	8	12	13	16
SW-100-4-0d3-trial2	100	200	0.0404	7	6	-	-	-	8	8	8	11	12	15
SW-100-4-0d3-trial3	100	200	0.0404	7	7	-	-	-	8	9	8	11	12	15
SW-100-5-0d1-trial1	100	200	0.0404	7	8	-	-	-	9	10	9	14	15	19
SW-100-5-0d1-trial2	100	200	0.0404	7	9	-	-	-	10	11	10	15	15	22
SW-100-5-0d1-trial3	100	200	0.0404	7	11	-	-	-	12	13	12	15	16	21
SW-100-5-0d2-trial1	100	200	0.0404	7	8	-	-	-	9	10	10	13	14	17
SW-100-5-0d2-trial2	100	200	0.0404	7	9	-	-	-	9	10	10	12	13	17
SW-100-5-0d2-trial3	100	200	0.0404	7	7	-	-	-	8	9	9	12	13	18
SW-100-5-0d3-trial1	100	200	0.0404	7	6	-	-	-	8	8	8	11	12	15
SW-100-5-0d3-trial2	100	200	0.0404	7	6	-	-	-	8	8	8	11	12	16
SW-100-5-0d3-trial3	100	200	0.0404	7	6	-	-	-	8	8	8	11	12	15
SW-100-6-0d1-trial1	100	300	0.0606	7	5	-	-	-	7	8	8	12	13	16
SW-100-6-0d1-trial2	100	300	0.0606	7	6	-	-	-	8	9	8	12	13	16
SW-100-6-0d1-trial3	100	300	0.0606	7	6	-	-	-	7	8	8	12	14	17
SW-100-6-0d2-trial1	100	300	0.0606	7	6	-	-	-	7	8	7	11	13	15
SW-100-6-0d2-trial2	100	300	0.0606	7	4	-	-	-	7	8	7	10	12	14
SW-100-6-0d2-trial3	100	300	0.0606	7	4	-	-	-	7	8	7	10	12	15
SW-100-6-0d3-trial1	100	300	0.0606	7	4	-	-	-	7	8	7	10	11	14
SW-100-6-0d3-trial2	100	300	0.0606	7	5	-	-	-	7	8	7	10	11	13
SW-100-6-0d3-trial3	100	300	0.0606	7	5	-	-	-	7	8	7	10	11	14

Outline

1. Introduction
2. Preliminaries and Literature Review
3. Optimal broadcasting in Fully-Connected Trees
4. A Broadcasting Heuristic for Hypercube of Trees
5. Fully-adaptive Model for Broadcasting with Universal Lists
6. Non-adaptive Broadcasting
7. Broadcast Graphs under the Fully-adaptive Model
8. HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm
9. Conclusion and Future Works

Conclusion

- For classical model:
    - ◇ An exact algorithm for FCT_n,
    - ◇ A heuristic for HT_k.

Conclusion

- For classical model:
  - ◇ An exact algorithm for FCT_n,
  - ◇ A heuristic for HT_k.
- Suggesting fully-adaptive model:
  - ◇ mbg's for $n \leq 10$,
  - ◇ bg's for $11 \leq n \leq 14$,
  - ◇ The first infinite family of bg's under universal lists model,
  - ◇ Exact value of $B_{fa}(G)$ for: trees, grids, hypercubes, cube connected cycles.
  - ◇ Upper bound on $B_{fa}(G)$ for tori.

Conclusion

- For non-adaptive model,
  - Exact value of $B_{na}(G)$ for: k-ary trees, binomial trees,
  - Upper bound on $B_{na}(G)$ for complete bipartite graph,
  - A general upper bound for trees.

Conclusion

- For non-adaptive model,
  - Exact value of $B_{na}(G)$ for: k-ary trees, binomial trees,
  - Upper bound on $B_{na}(G)$ for complete bipartite graph,
  - A general upper bound for trees.
- `HUB-GA`
  - The first heuristic for the problem of broadcasting with universal lists.

Future Works

- Chapter 3: close the gap between the obvious lower bound $\Omega(|V|)$ and the current algorithm $O(|V| \log \log n)$.

Future Works

- Chapter 3: close the gap between the obvious lower bound $\Omega(|V|)$ and the current algorithm $O(|V| \log \log n)$.
- Chapter 4: replace hypercube with other graphs with known broadcast time,

Future Works

- Chapter 3: close the gap between the obvious lower bound $\Omega(|V|)$ and the current algorithm $O(|V| \log \log n)$.
- Chapter 4: replace hypercube with other graphs with known broadcast time,
- Chapter 5:
  - Broadcast time of many networks are still unknown under the fully-adaptive model,
  - Improving the current upper bound on complete graphs,
  - Studying the widest margin between a graph's classical and fully-adaptive broadcast time on n vertices
  - Studying the hardness of the problem.

Future Works - cont.

- Chapter 6: broadcast time of different interconnection networks under non adaptive model.

Future Works - cont.

- Chapter 6: broadcast time of different interconnection networks under non adaptive model.
- Chapter 7:
  - Finding *mbg*'s and *bg*'s for greater values of n,
  - Is there any value of n, for which $B^{(cl)}(n) < B^{(fa)}(n)$?
  - Defining *bg*'s for adaptive and non-adaptive models (where the reachability of the obvious lower bound of $\lceil \log n \rceil$ is questionable).

Future Works - cont.

- Chapter 6: broadcast time of different interconnection networks under non adaptive model.
- Chapter 7:
  - Finding *mbg*'s and *bg*'s for greater values of n,
  - Is there any value of n, for which $B^{(cl)}(n) < B^{(fa)}(n)$?
  - Defining *bg*'s for adaptive and non-adaptive models (where the reachability of the obvious lower bound of $\lceil \log n \rceil$ is questionable).
- Chapter 8:
  - Experiments on more data,
  - Trying different algorithms such as Ant Colony or particle swarm optimization,
  - Proposing a similar approach for minimizing $B_{cl}(G)$, not for a particular vertex!

Publications

- Chapter 3:
  - **Gholami, S.**, Harutyunyan, H. A., & Maraachlian, E. (2022). Optimal Broadcasting in Fully Connected Trees. *Journal of Interconnection Networks*, 2150037.
- Chapter 4:
  - **Gholami, S.**, & Harutyunyan, H. A. (2021). A Broadcasting Heuristic for Hypercube of Trees. In *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 0355–0361).
- Chapter 5:
  - **Gholami, S.**, & Harutyunyan, H. A. (2022b). Fully-adaptive Model for Broadcasting with Universal Lists. In *24th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*.

Publications - cont.

- Chapter 6:
  - **Gholami, S.**, & Harutyunyan, H. A. (2022d). A Note on Non-adaptive Broadcasting with Universal Lists. *Special issue on Graph and Combinatorial Optimization for Big Data Intelligence with Parallel Processing, Parallel Processing Letters* (Under Review).
- Chapter 7:
  - **Gholami, S.**, & Harutyunyan, H. A. (2022a). Broadcast Graphs with Nodes of Limited Memory. In *13th International Conference on Complex Networks (CompleNet)*.
- Chapter 8:
  - **Gholami, S.**, & Harutyunyan, H. A. (2022c). HUB-GA: A Heuristic for Universal lists Broadcasting using Genetic Algorithm. *Journal of Communications and Networks* (Accepted).

Publications - cont.

- In collaboration with other researchers:
  - ◇ Bakhtar, S., **Gholami, S.**, & Harutyunyan, H. A. (2020). A new metric to evaluate communities in social networks using geodesic distance. In *International Conference on Computational Data and Social Networks (CSoNet)* (pp. 202–216).
  - ◇ **Gholami, S.**, Saghiri, A. M., Vahidipour, S., & Meybodi, M. (2021). HLA: a novel hybrid model based on fixed structure and variable structure learning automata. *Journal of Experimental & Theoretical Artificial Intelligence*, 1–26.

Thanks a bunch!